Grid Computing: Topology-Aware, Peer-to-Peer, Power-Aware, and Embedded Web Services

Craig A. Lee, lee@aero.org
The Aerospace Corporation
(A non-profit, federally funded research & development corp.)

Seventh Annual High Performance Embedded Computing Workshop
MIT Lincoln Laboratory, Lexington, Massachusetts, September 22-25, 2003

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **20 AUG 2004**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**
4. TITLE AND SUBTITLE **Grid Computing: Topology-Aware, Peer-to-Peer, Power-Aware, and Embedded Web Services**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **The Aerospace Corporation**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM001694, HPEC-6-Vol 1 ESC-TR-2003-081; High Performance Embedded Computing (HPEC) Workshop (7th)., The original document contains color images.

14. ABSTRACT

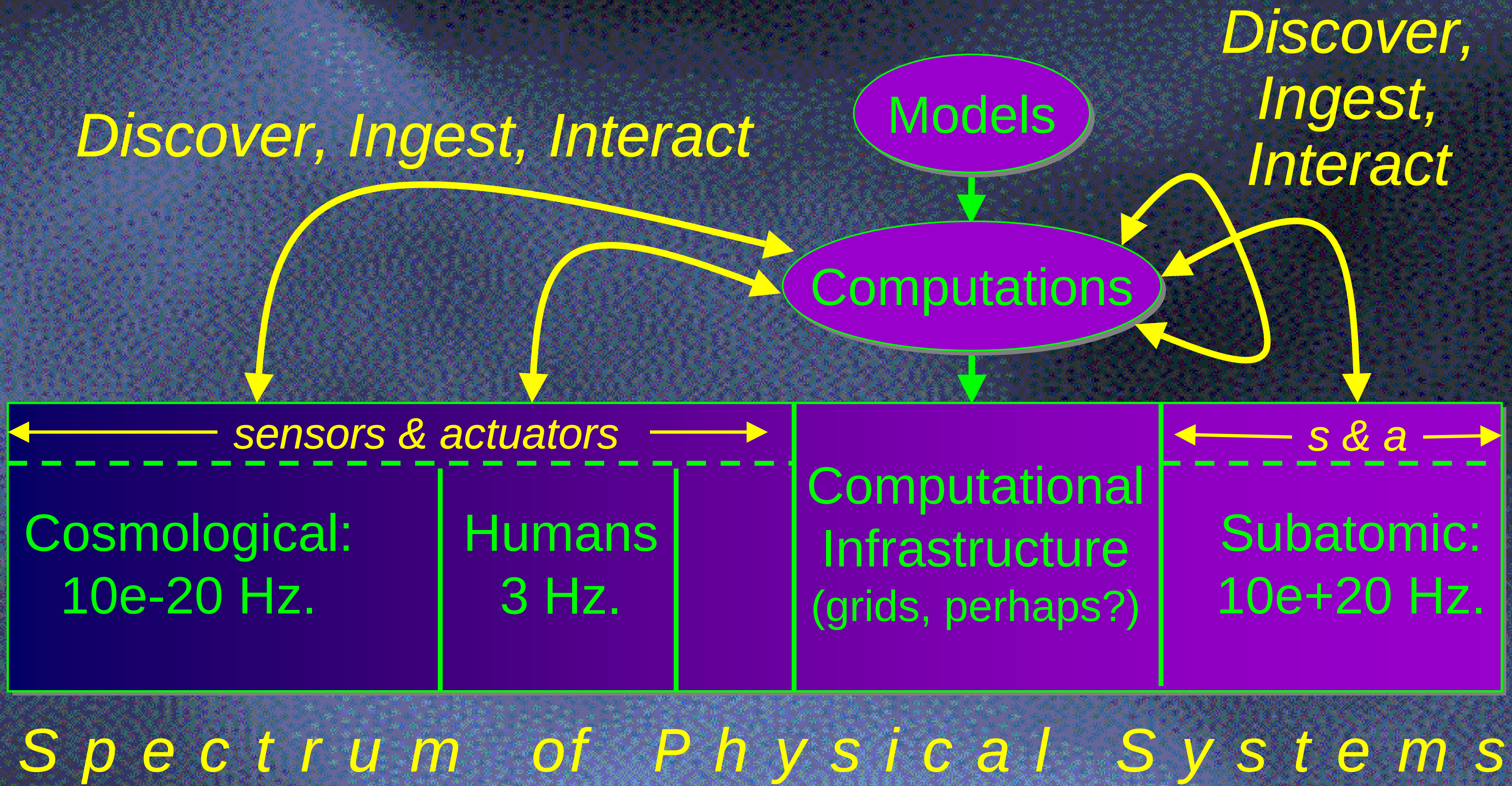
15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**40**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

A DDDAS Model

(Dynamic, Data-Driven Application Systems)



A DDDAS Example: Forest Fires

Kirk Complex Fire. U.S.F.S. photo

DDDAS Issues *Grid Issues*

- Information Metadata Schemas
- Information and Resource Discovery
- Scheduling & Co-Scheduling
- Cycles, Memory, Bandwidth, Latency
- Wired, Mobile, & Ad Hoc Communication
- Event Services, Messaging Services
- Timeliness, Control Feedback
- Performance Monitoring
- Fault Tolerance
- Security

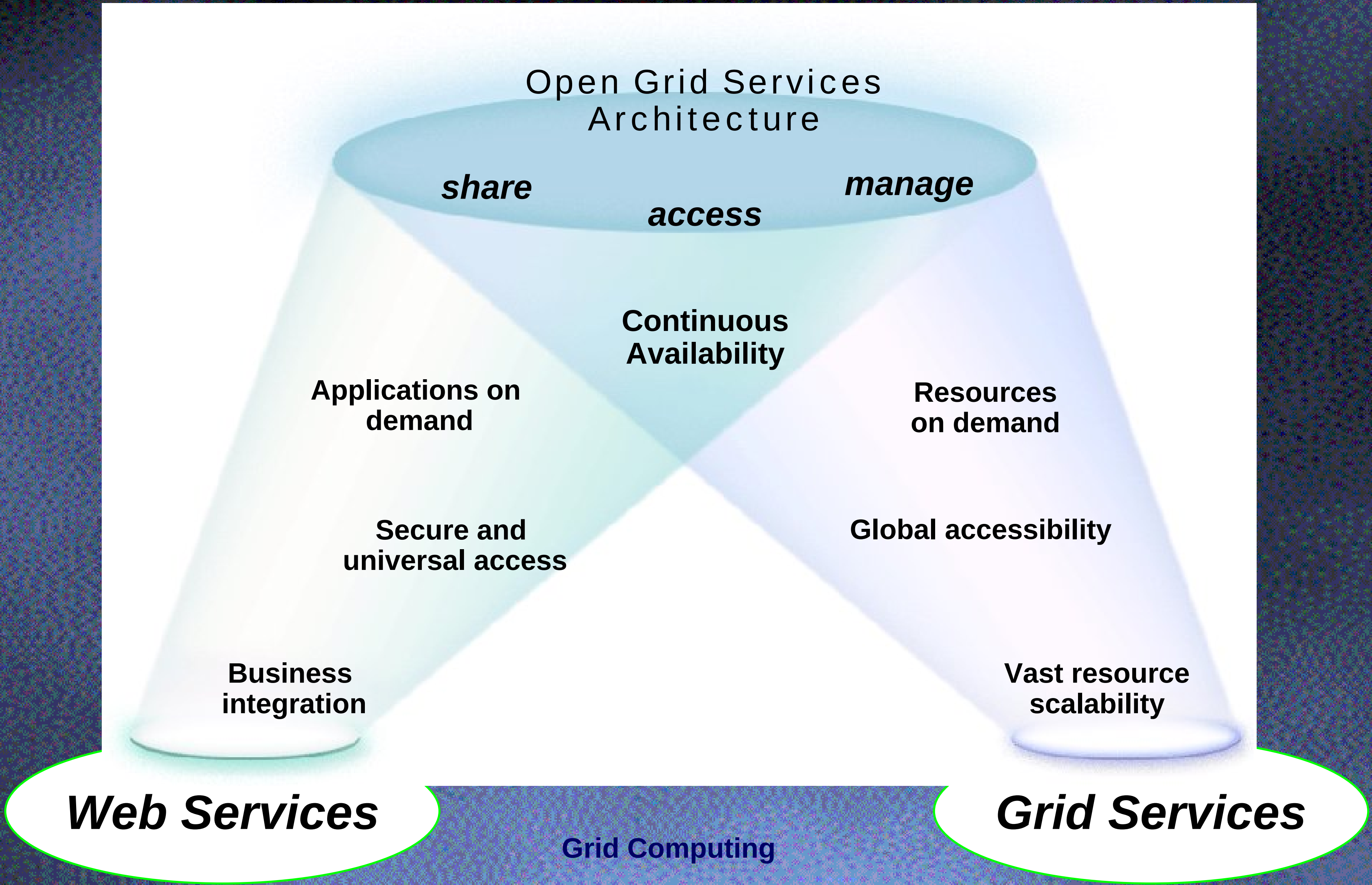
Grid Computing

- *What is it?*
  - Distributed, networked computing
  - Heterogeneous, distributed, virtual supercomputing
  - "Information Power Grid" is analogous to the Electrical Power Grid -- It's always there & available
- Flexible integration of all manner of resources
  - Time-shared and space-shared machines of all sizes
  - Specialized software and hardware resources
    - e.g., X-ray sources, satellite downlink, very large databases
- *An Enabling Technology*
  - Cost-effective aggregation of compute power to achieve compute power not possible any other way
  - Virtual Organizations

Open Grid Services Architecture

- Service Architecture comprised of:
  - *Persistent Services* (typically a few)
  - *Transient Services* (potentially many)
  - All services adhere to specified Grid service interfaces and behaviors
    - Reliable invocation, lifetime management, discovery, authorization, notification, upgradeability, concurrency, manageability
- Interfaces for managing Grid service instances
  - *Factory, registry, discovery, lifetime*, etc.
- ➔ Reliable, secure mgmt of distributed state
    - Full details available from *www.globus.org/ogsa*

OGSA : A Generalization of Web Services



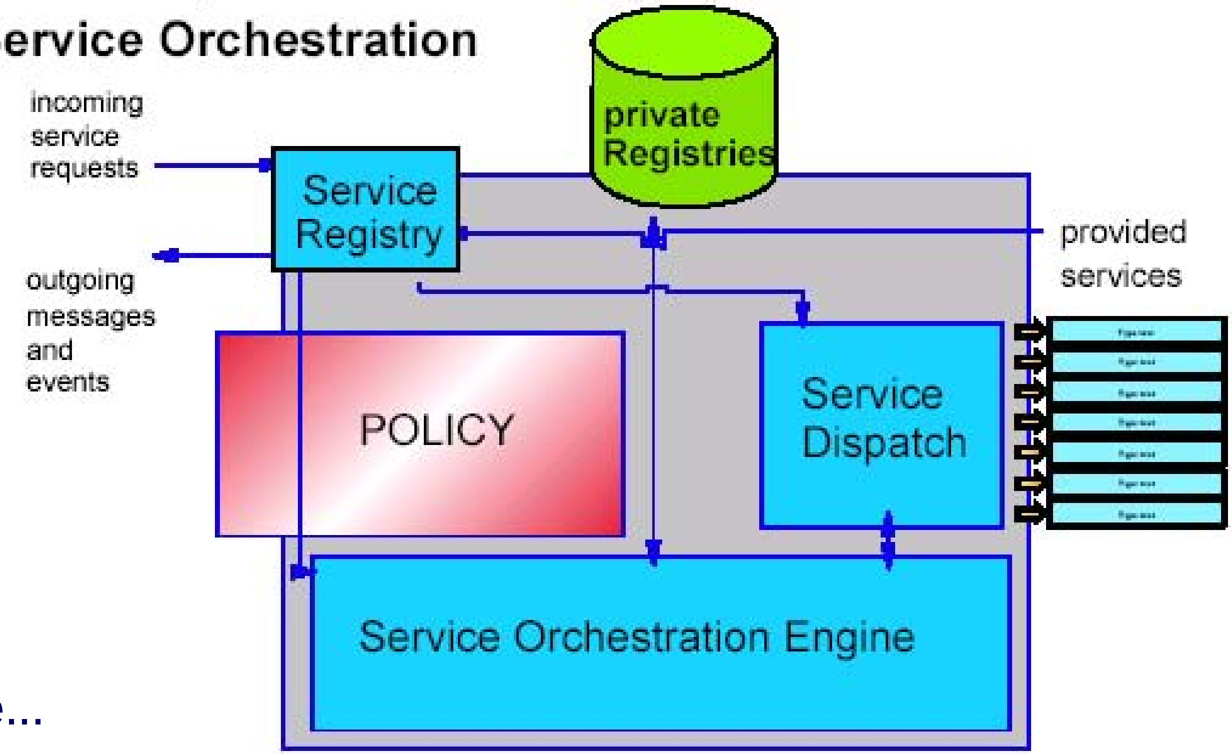
Web Services (W3C)

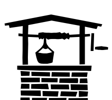
- Increasingly popular standards-based framework for accessing network applications
  - World-Wide Web Consortium (W3C) Standardization
    - Microsoft, IBM, Sun, others
  - WSDL: Web Services Definition Language
    - Interface definition Language for Web Services
  - SOAP: Simple Object Access Protocol
    - XML-based RPC protocol; common WSDL target
  - WS – Inspection
    - Conventions for locating service descriptions
  - UDDI: Universal Description, Discovery & Integration
    - Directory for Web Services

OGSA: A type of *Component Architecture*

Service Domains: Distributed System Components

- Service Registration and Collection
- Service Routing and Selection
- Service Interoperation and Transformation
- Flexible Service Composition
- Autonomic Service Orchestration





From the source...

How to Make All of This Accessible for Non-Specialists Using Existing Traditional Programming Tools?

- *GridRPC*
  - Remote Procedure Call extended for grid environments using grid services
- Established programming paradigm
  - *Low barrier to adoption*
- *Implementable on top of OGSA*
- GGF GridRPC Working Group
  - *http://graal.ens-lyon.fr/GridRPC*
- Motivated by *Network-Enabled Services*
  - e.g., NetSolve, Ninf-G, DIET

GridRPC Prototypes

NetSolve

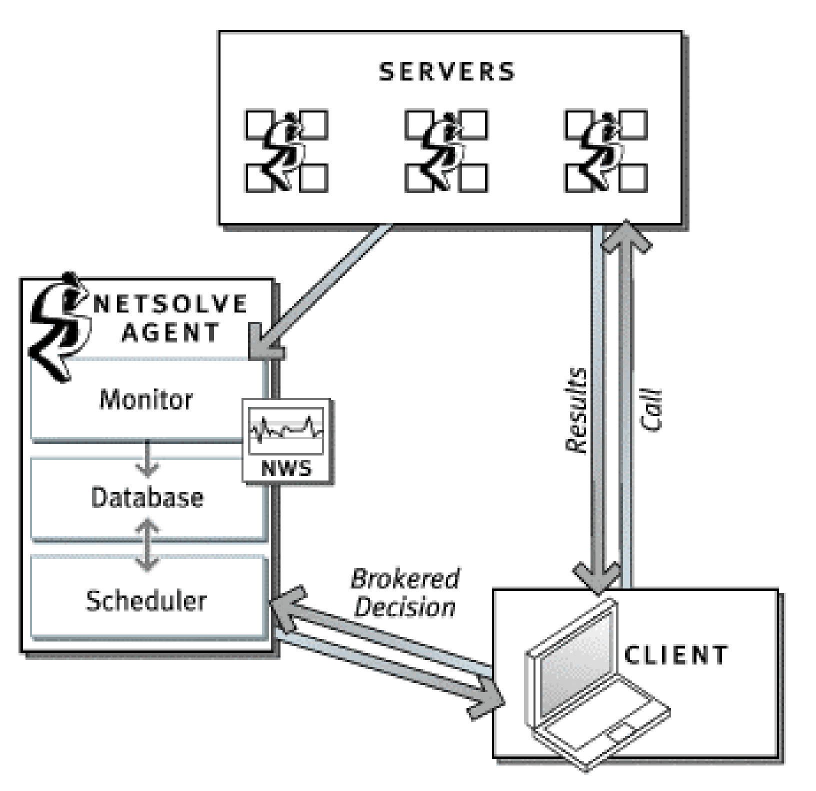
J. Dongarra

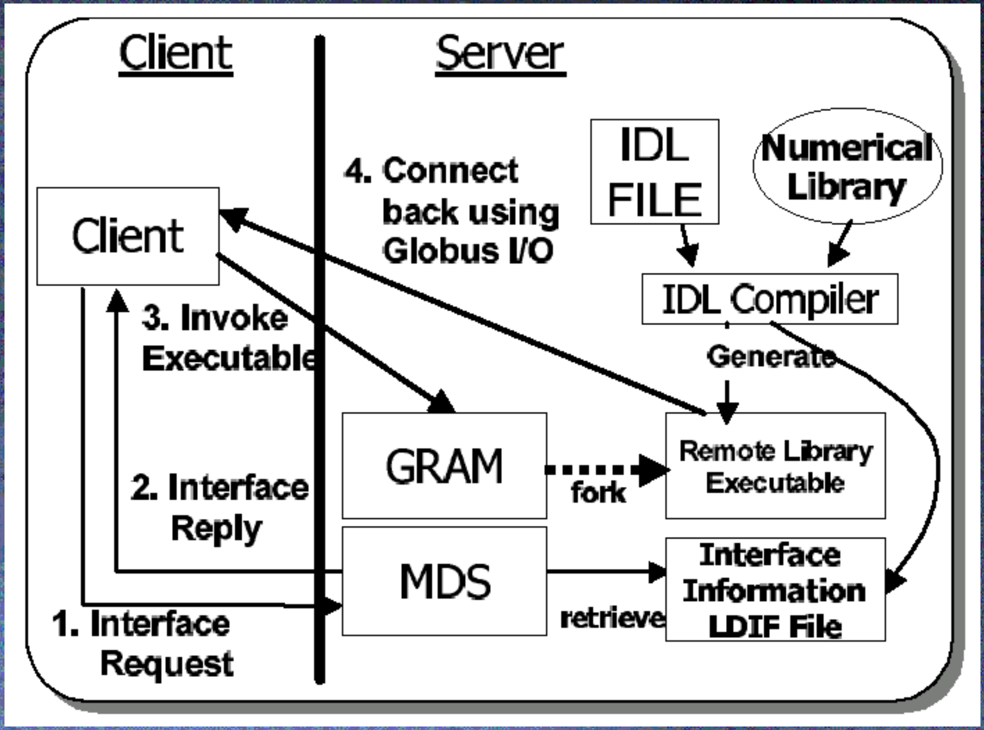
U. Tenn, Knoxville

Ninf-G

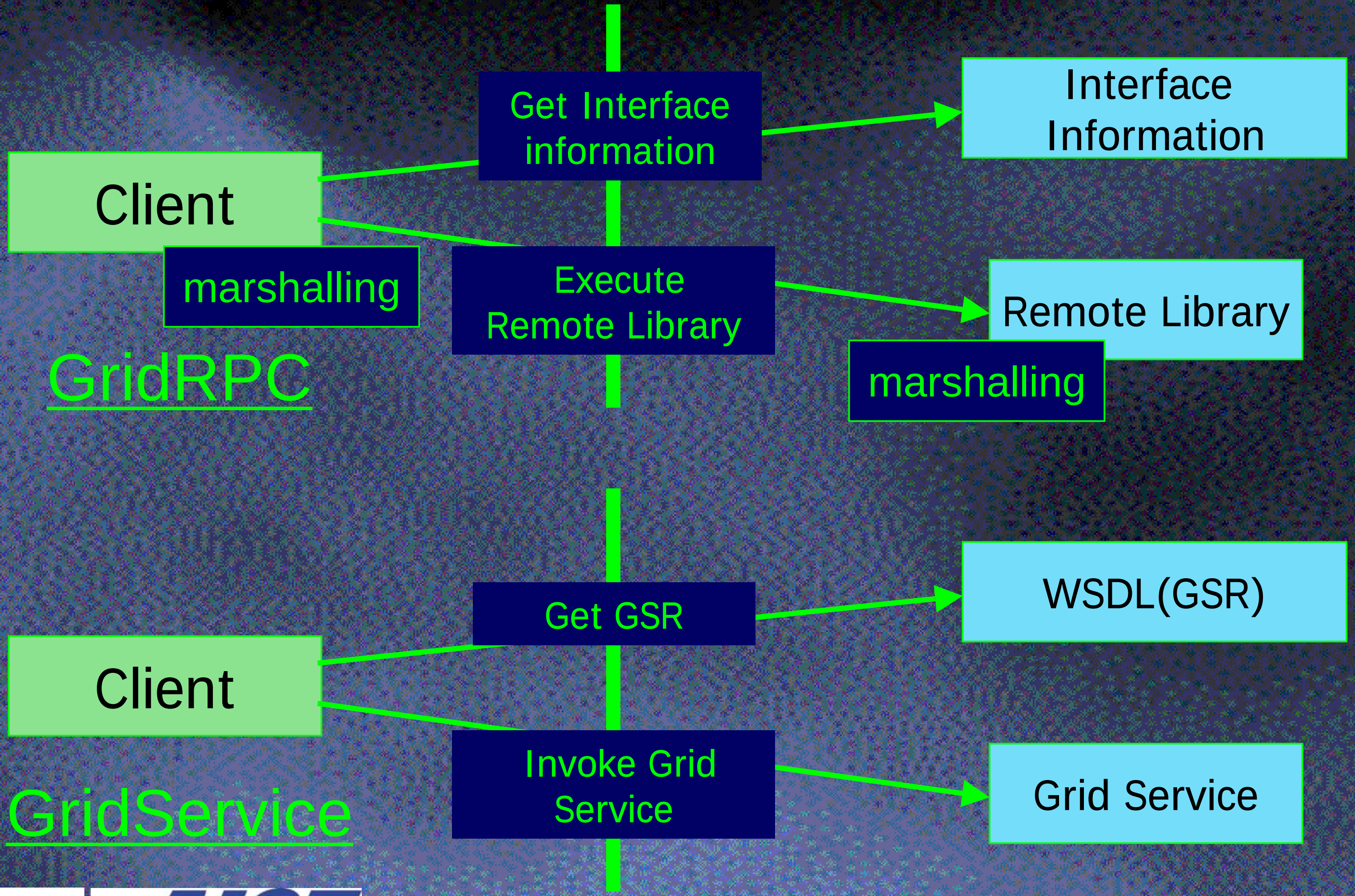
S. Matsuoka

Tokyo Inst. of Tech.





Comparing GridRPC and Grid Services



- *Looks quite similar, but …*

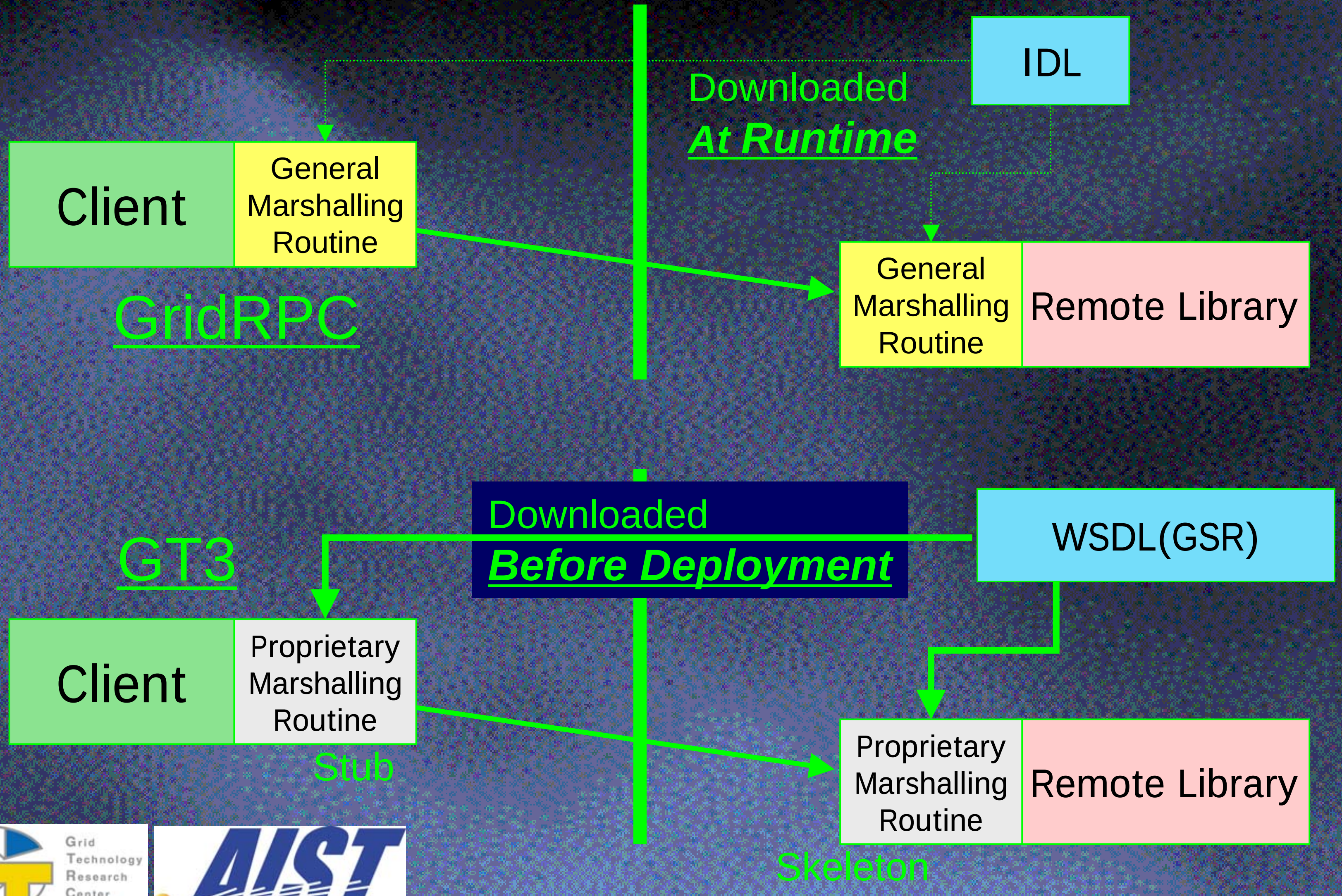
From the source…

.... turns out to be not so easy

- GT3 (Apache Axis) does not interpret WSDL at runtime
  - Statically interpret WSDL and generate Java Proxy class for the client
    - Data Marshalling is hardcoded in the proxy class
    - Client programmer has to download the proxy class before writing his/her code
  - WSDL downloaded at runtime is used just to get the location of the service

From the source...

GridRPC and Grid Service (GT3)



From the source...

Impact for Programming with Grid Services

- GridRPC: Simple Client-side Programming & Mgmt
  - No client-side stub programming or IDL management
- Dynamic WSDL run-time interpretation needed!
  - Without it, GSH-GSR Resolution is limited
  - Lack impacts implementability of GridRPC on top of OGSA
- Very Late Binding is necessary
- Alternate Approach: *Representational State Transfer*
  - www.ics.uci.edu/~taylor/documents/2002-REST-TOIT.pdf
- RESTful interactions are stateless
  - Each request contains all necessary information for connector and service to understand request
  - Could be represented as XML document
- "Smart Run-time" could cache known services based on stable availability, usage patterns and "compile" them in
  - Ultimate trade-off between what is reliably static and known a priori, and what must be dynamic and discovered at run-time

RESTful Namespaces

- URL as a six-tuple:

  protocol://network_loc/path;args?query#fragment

  path = service_name/service_instance
- Service provider is master of its namespace
  – Manages both persistent and transient naming
- Well-known naming convention possible

  …/path/status

  …/path/log

  …/path/debug

  …/path/cancel

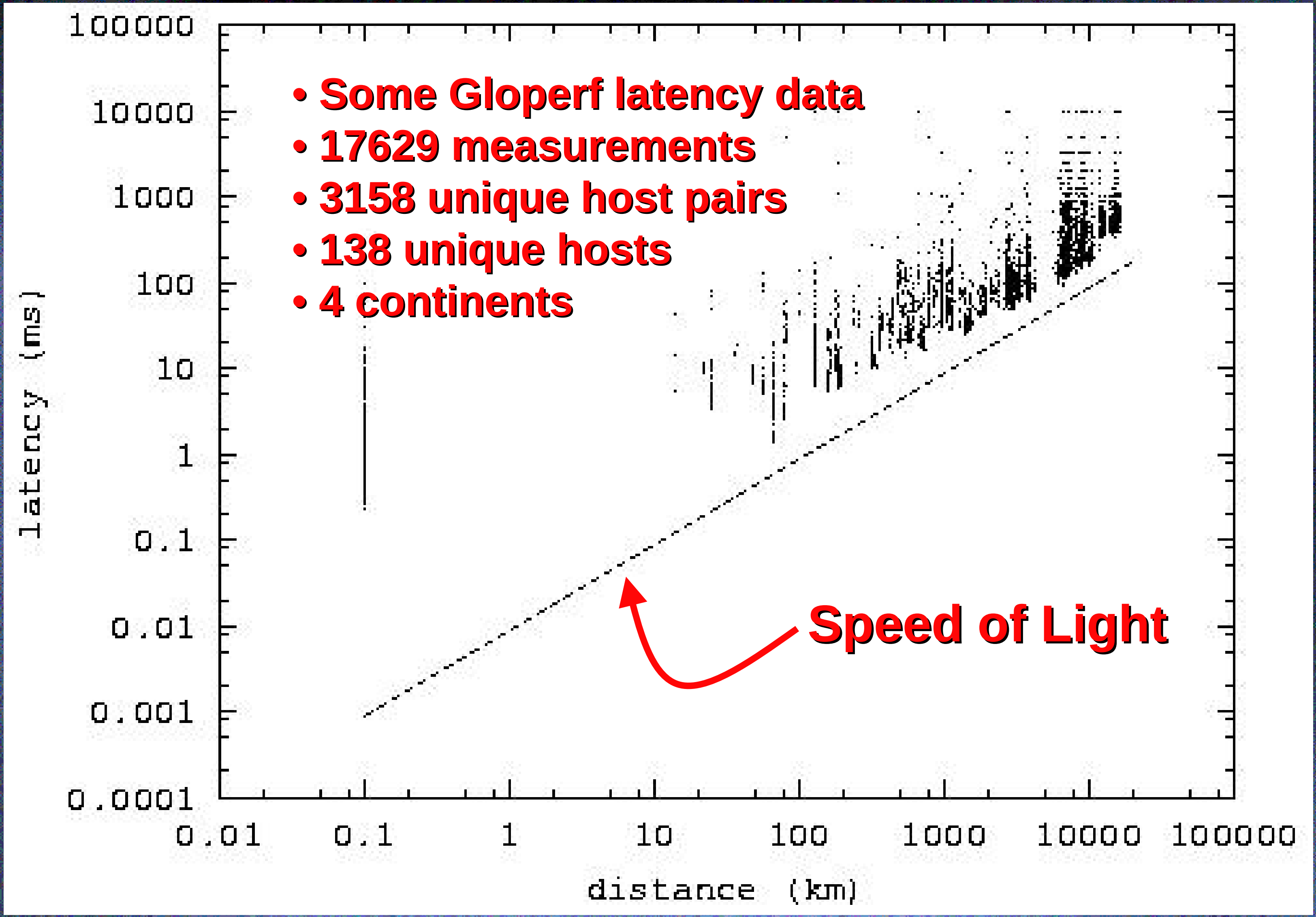
  …/path/result

GridRPC function handles, session IDs, and data handles become URLs

Now, What about Performance for Wide-Area Grid Computations?

- Grids promise an unprecedented degree of distributed computing
  - A fabric of network-connected sites and resources
- As processors and networks get faster, grid computations will become increasingly *latency-sensitive*

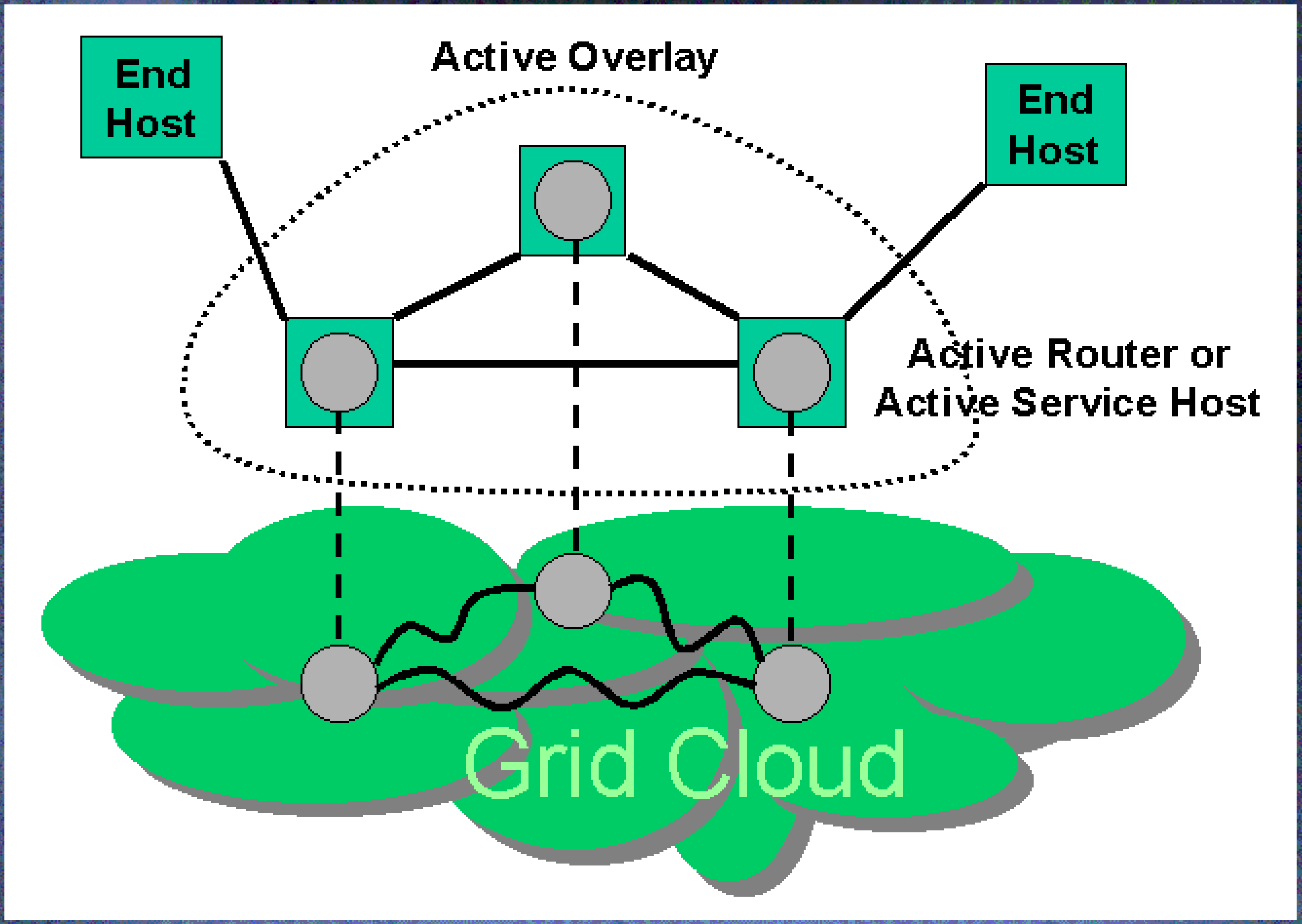
How Latency Sensitive Is It?



Why Topology-Aware Communication Services?

- The network topology connecting these sites and resources can be exploited
  - Improve performance
  - Enable new functionality
- Topology-awareness will become *essential*

Topo-Aware Comm Services Can Be Similar to an Overlay



Many Types of Communication Services Improved or Enabled

- Augmented Semantics
  - Caching (web caching), filtering, compression, encryption, quality of service, data-transcoding, etc.
- Collective Operations
  - Accomplished “in the network” rather than using point-to-point msgs across the diameter of the grid
- Communication Scope
  - *Named topologies* can denote a communication *scope* to limit problem size and improve performance
- Content and Policy-based networking
  - Publish/subscribe, interest management, event services, tuple spaces, quality of service

A Collective Op Case Study: Time Mgmt in Dist Simulation

- Time Management enables *temporal causality* to be enforced in Distributed Simulations
- Typically enforced via a *Lower Bound Time Stamp* (LBTS) algorithm
- *Topology-Aware Communication is a natural*
  - Eliminates point-to-point communication
  - Increase performance for LBTS, the key TM algorithm
- *Distinguished Root Node Algorithm* developed as a topology-aware time management service
  - Relies on a tree from end-hosts to a distinguished root node in the network
  - Instance of the *Distributed Termination Detection* problem

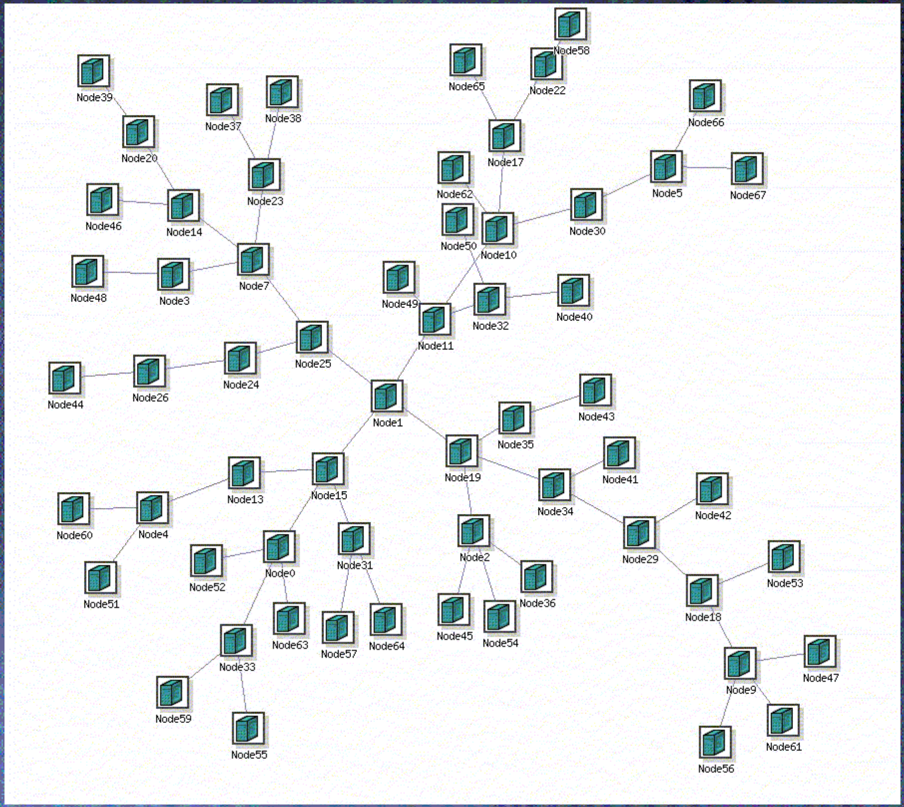
Metropolitan Testbed for Distinguished Root Node Algorithm

Too Small for Convincing Results!

Eric Coe made this work!

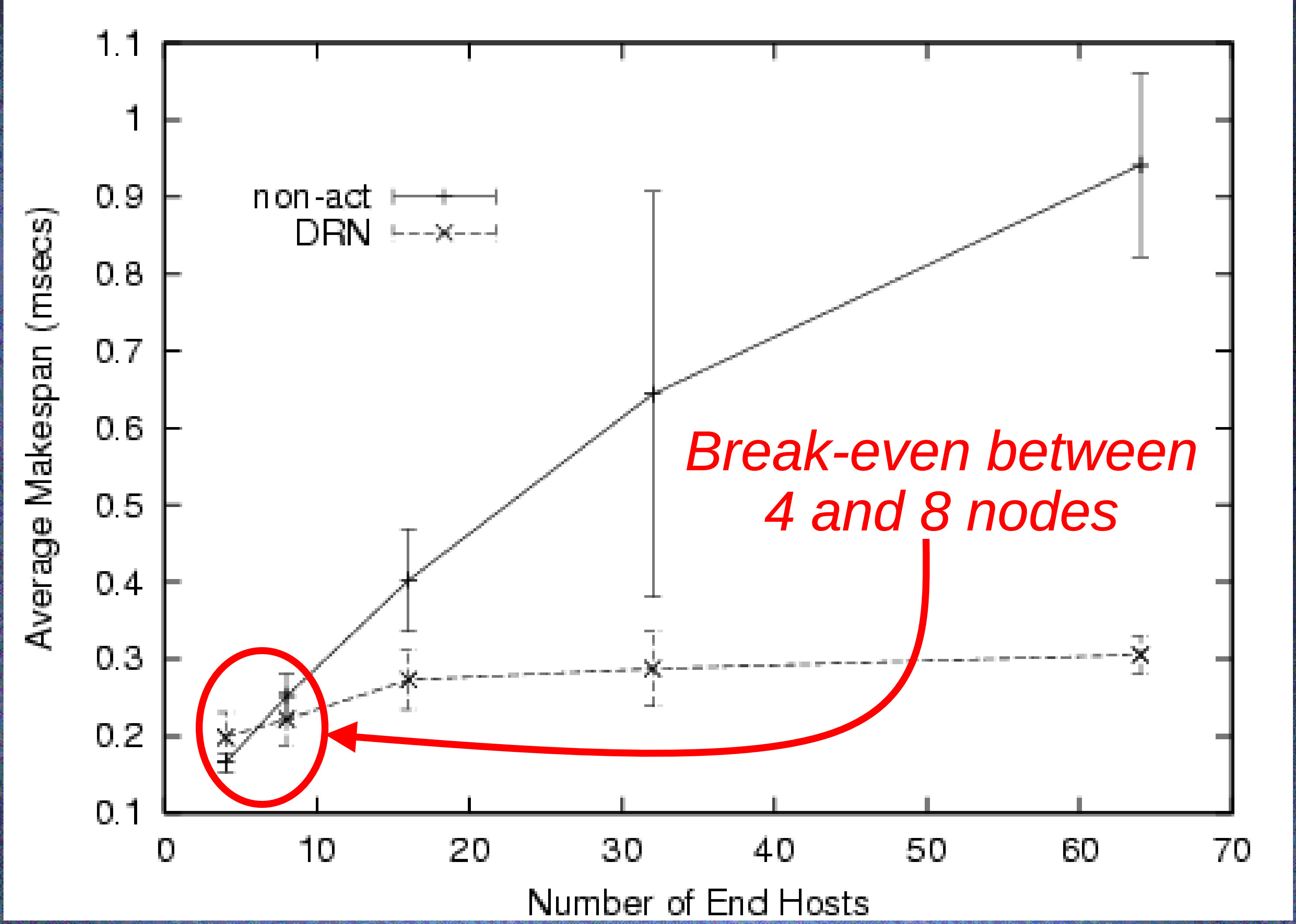
How to Run More Realistic Cases? *EmuLab*

- Network emulation cluster at Utah
  - www.emulab.net
- DRN and traditional, point-to-point algorithms compared on larger topologies
- Topologies run with up to 98 nodes
- Eric made this work, too!



Example: 32 end-hosts, 29 routers

LBTS Makespan on EmuLab (ms)



Content-Based Networking

- Content-Based Routing
  - Message-Passing with Associative Addressing
  - Requires an associative matching operation
- A fundamental and powerful capability
  - Enables a number of very useful capabilities and services
  - Event services, resource discovery, coordination programming models
- But notoriously expensive to implement
  - How can matching be done efficiently in a wide-area grid env?
- *Can users and apps find a "sweet-spot" where content-based routing is constrained enough to be practical and provide capabilities that can't be accomplished any other way?*
  - *Scale of deployability*

Example: Scalability of Distributed Simulation

What We Have...

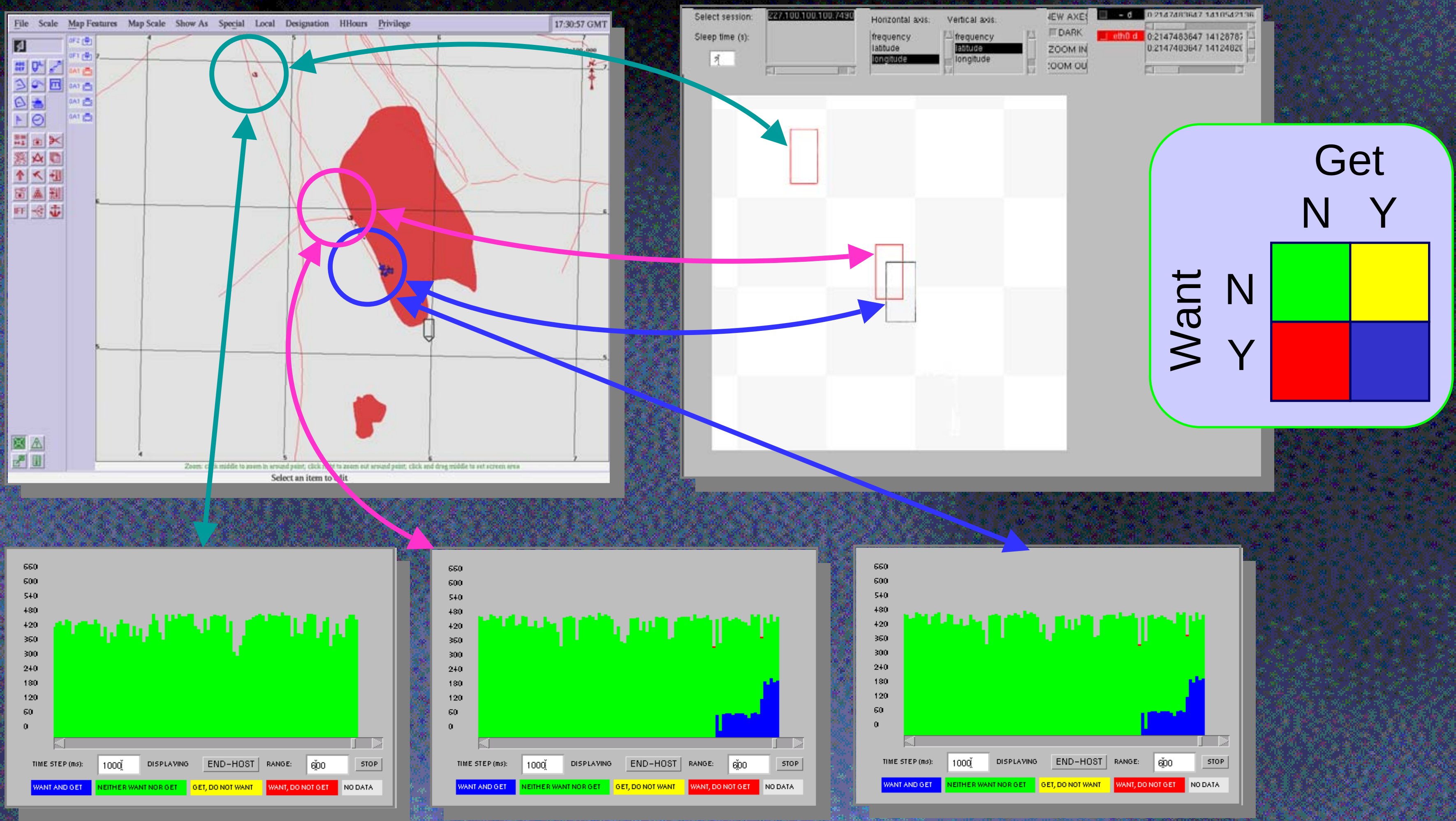
- Multicasting to improve send-side scalability for one-to-many delivery of simulated entity state updates
- Receiver and network overload from delivery of updates from *far* more entities than wanted or needed locally

What We Want...

- Means for subscribing to, and receiving only state updates that are needed and relevant
  -- ***content-based routing***

DARPA Active Networks Demo, Dec. 2000, Zabele, Braden, Murphy, Lee

Tank/Jet Fighter Engagement



Red Tank Platoon B

Red Tank Platoon A

Blue Airstrike

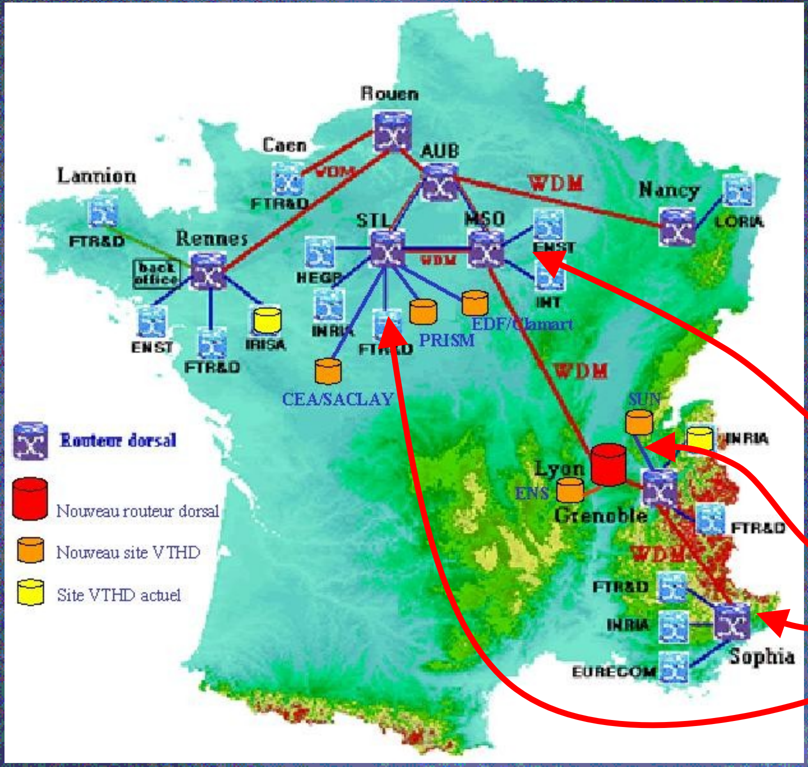
DARPA Active Networks Demo, Dec. 2000, Zabele, Braden, Murphy, Lee

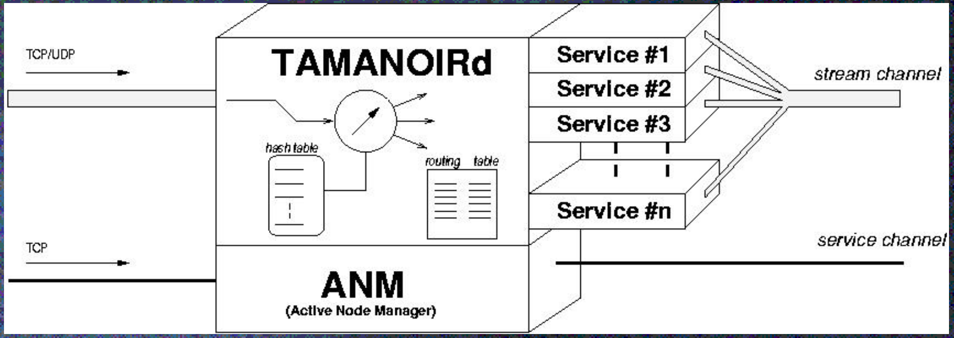
How Will Much of This Be Managed?

- Implementation Approaches:
  - Explicit Network of Servers
  - Active Networks
  - *Peer-to-Peer Middleware*

An Active Networks Approach: e-Toile et Tamanoir

French national grid project with Tamanoir daemons at major sites





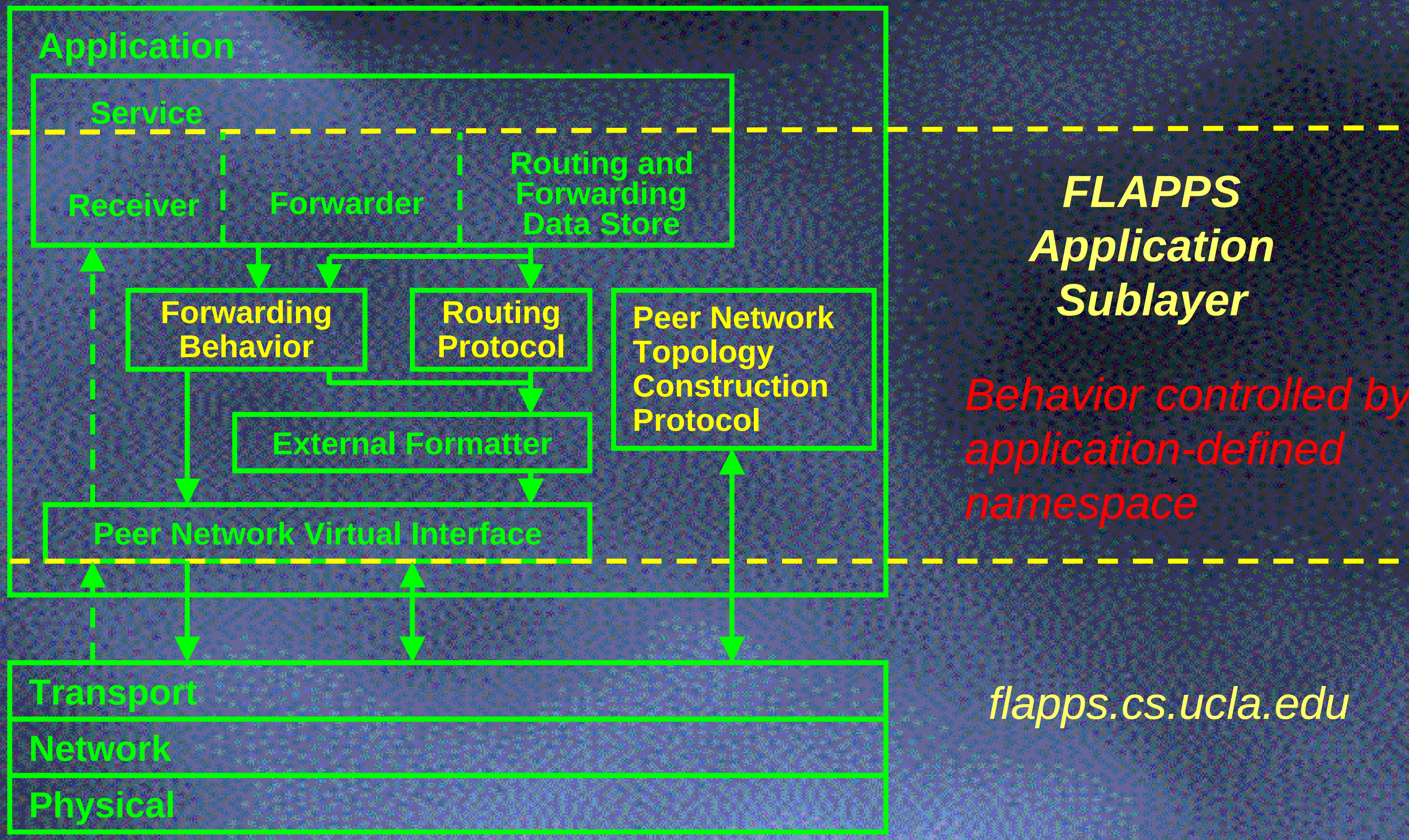
www.urec.cnrs.fr/etoile
www.ens-lyon.fr/~jpgelas/TAMANOIR

Host services such as:

- Internet Backplane Data Depots
- Reliable Multicast Repair
- Active Quality of Service

A Peer-to-Peer Approach: FLAPPS

(Forwarding Layer for Application-level Peer-to-Peer Services)

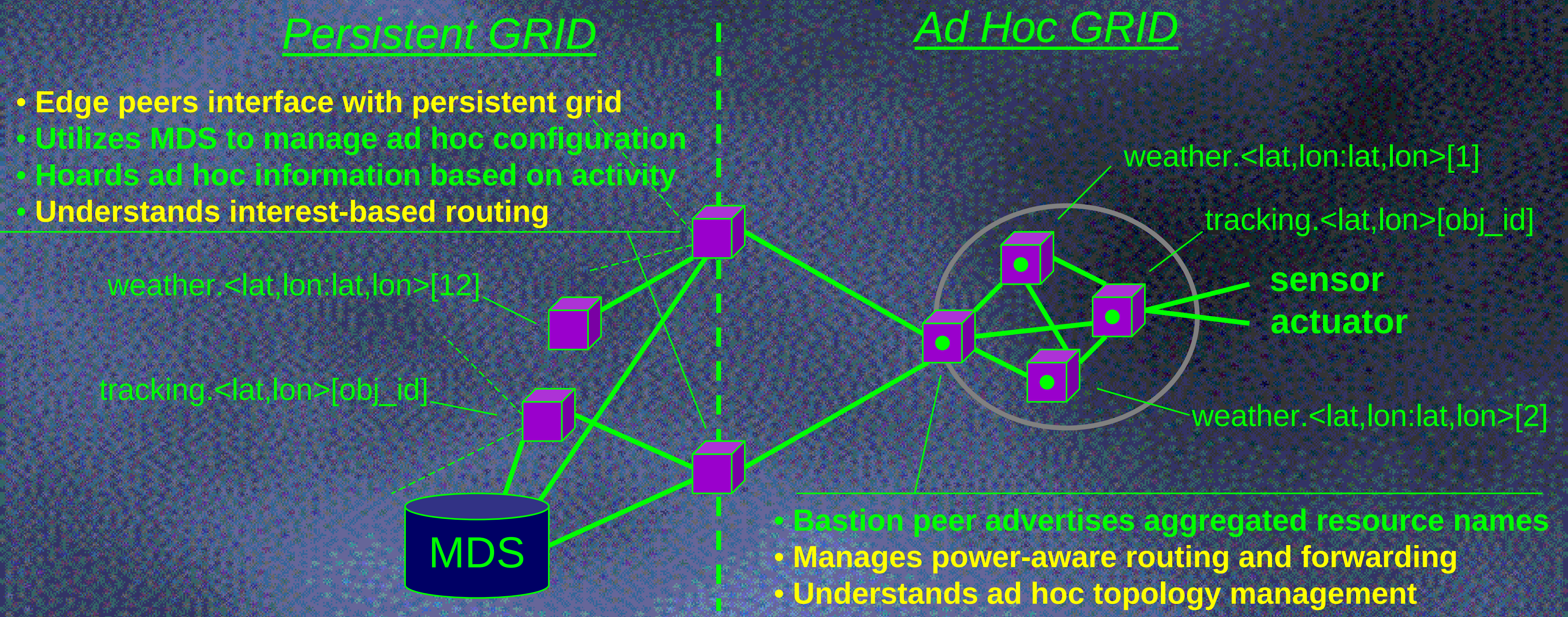


FLAPPS Application Sublayer

Behavior controlled by application-defined namespace

flapps.cs.ucla.edu

Interfacing Wired and Ad Hoc Grids with a FLAPPS Namespace



Namespace could be as general an XML DTD

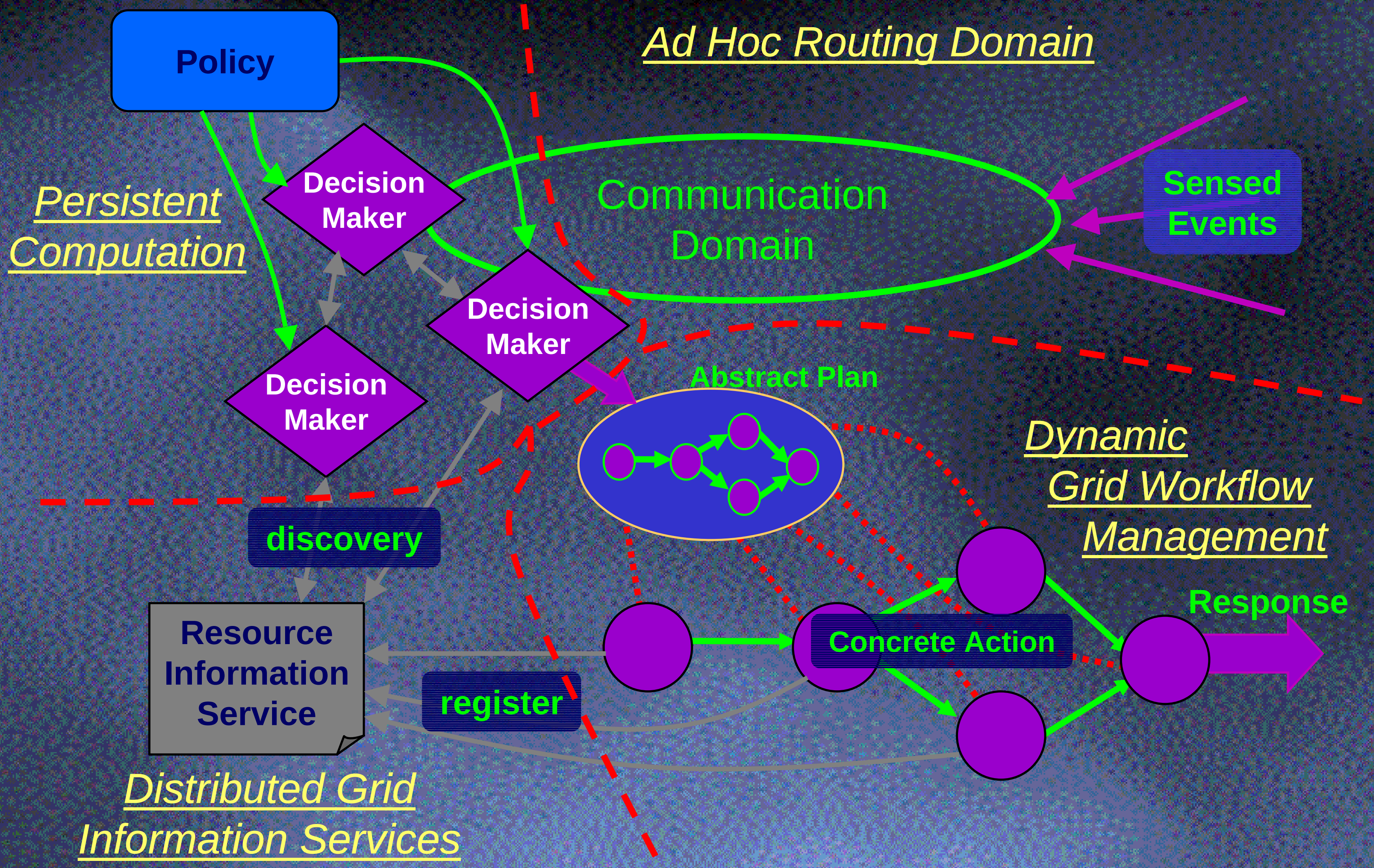
Issues Addressable…

- Embedded device capabilities will vary widely
  - Size, Power, Connectivity, etc.
- A well-known namespace convention and topology-aware P2P middleware layer will greatly facilitate the integration of all resources
  - Power-awareness and Power-oblivious
  - Compensate for lack of Mobile IP
    - e.g., in GSH-GSR resolution
  - “Smart” component connectors
- *Separation of low-level bit transmission from application-specific communication management*

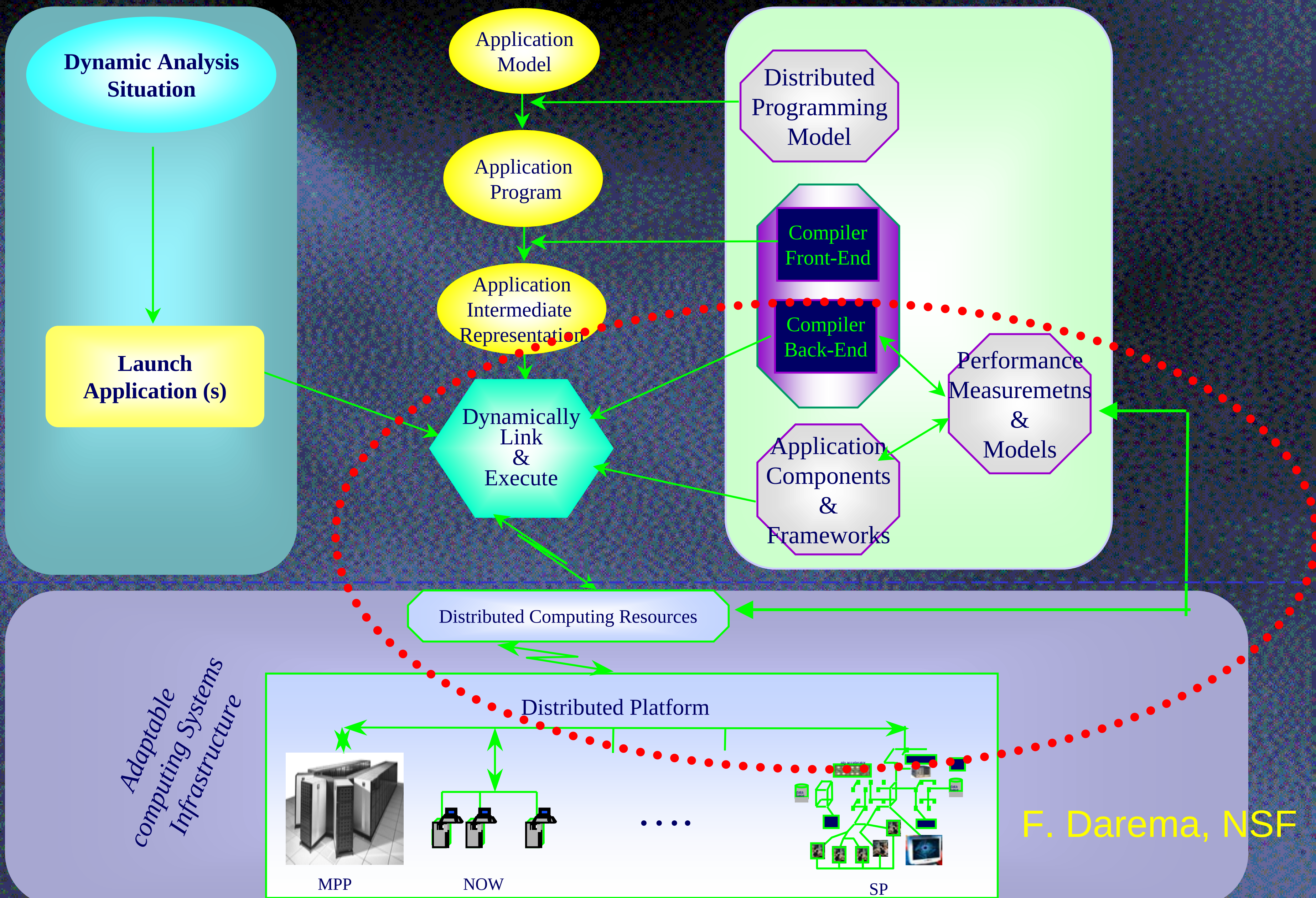
Other P2P Technologies

- Key-based/Distributed Hash Table Infrastructures
  - Pastry: Rice University
  - Chord: MIT
  - Content Addressable Networks (CAN): UC Berkeley
  - DHT emulation: FLAPPS peer service with binary identifier name space
  - FLAPPS message forwarding is explicit vs. transparent in DHTs
- JXTA: Sun Microsystems
  - "Network Pipe"-oriented P2P symmetric communication
  - JNGI: JXTA GRID workflow establishment project
  - JXTA's rendezvous nodes and peer group advertisements similar to topology construction

Return of the High-Level Concept



The NGS Program develops Technology for integrated feedback & control
Runtime Compiling System (RCS) and Dynamic Application Composition



F. Darema, NSF

Summary and Review!

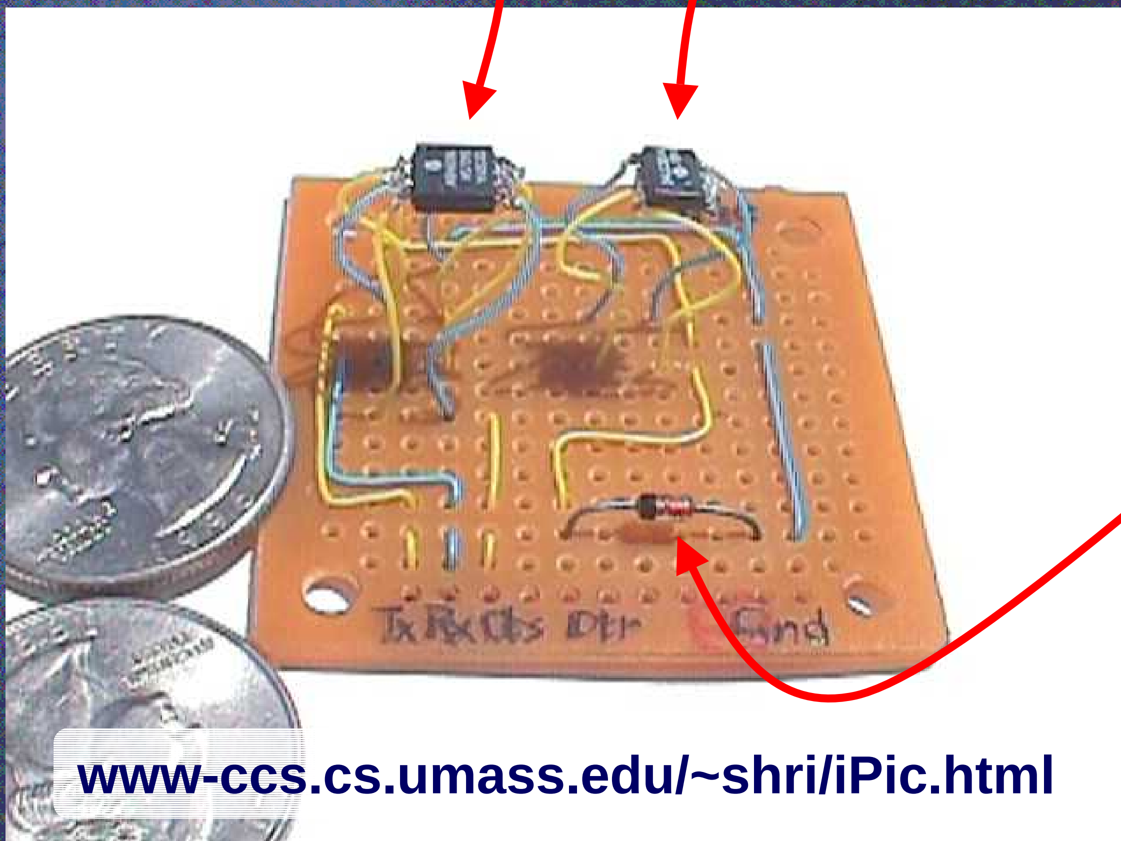
- Component "Web Service" Architectures with *well-known namespace conventions*
  - *GridRPC and OGSA are not the end of the story!*
- Topology-Aware Communication Services will become essential
  - Many important capabilities enabled
- Peer-to-Peer Systems will manage much of this
  - *Convergence of Grid and P2P!*
- Program meta-models w/ grid-aware "back-ends"
  - Coarse-grain, data-driven execution models
  - Optimistic or speculative execution models
- Mobile, Ad Hoc, Embedded grids are coming
  - *Complete DDDAS – How soon?*

iPic Web Server Hardware

PIC 12C509A Processor

24LC256 EEPROM

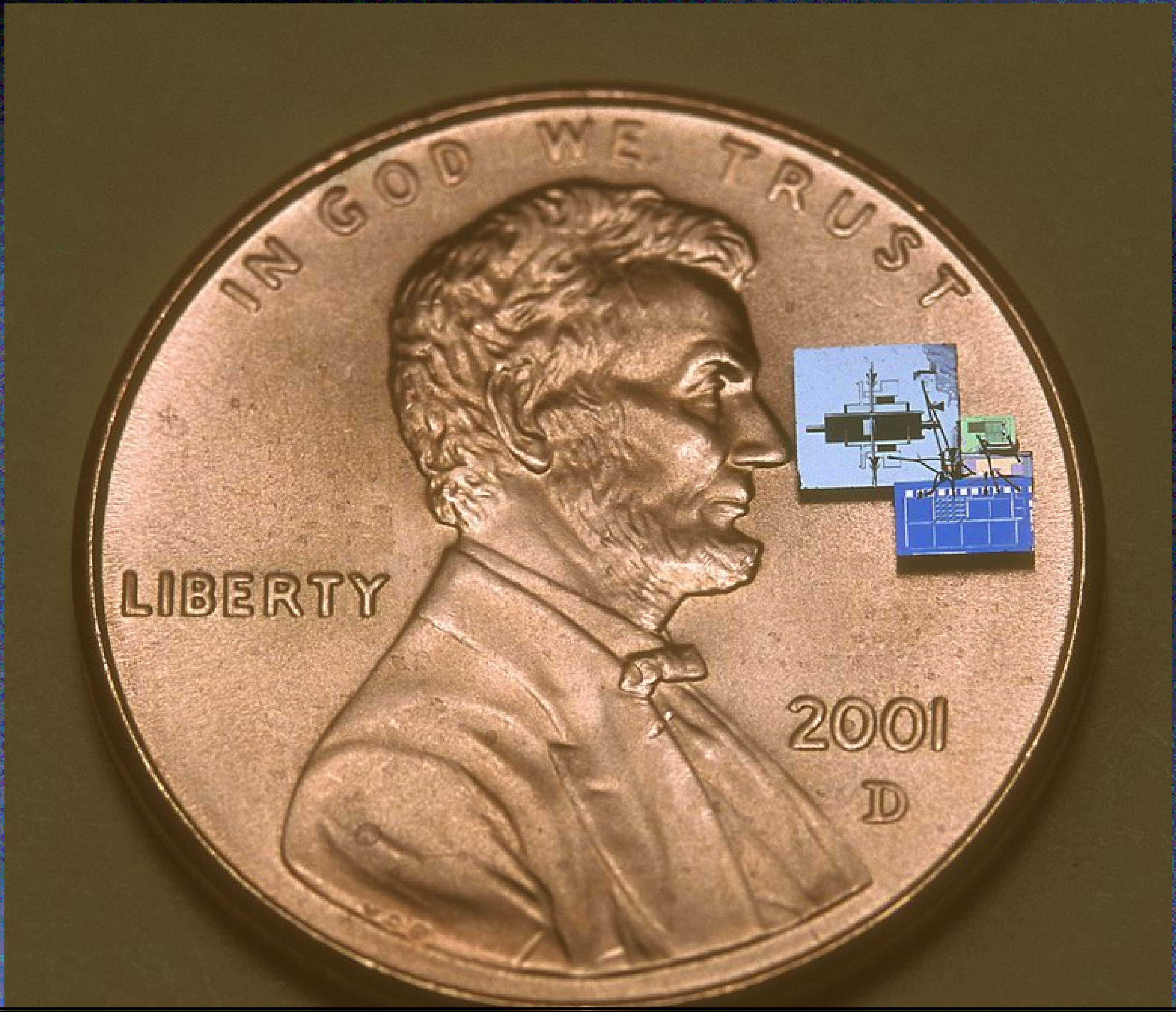
Power-supply regulator



www-ccs.cs.umass.edu/~shri/iPic.html

Even Smaller: Golem Dust

- Solar-powered
- Bi-dir comm
- Simple sensing
  - Acceleration
  - Ambient light



http://www-bsac.eecs.berkeley.edu/~warneke/SmartDust/index.html

The Future of Grid Deployment?

Questions?

lee@aero.org

http://robotics.eecs.berkeley.edu/~pister/SmartDust/BlowDust.htm